

PAPER • OPEN ACCESS

## Implementation of the nonlinear programming algorithm when optimizing the final turning mode

To cite this article: M G Galkin and A S Smagin 2020 *IOP Conf. Ser.: Mater. Sci. Eng.* **862** 032009

View the [article online](#) for updates and enhancements.

**ECS**  
The Electrochemical Society  
THE KOREAN ELECTROCHEMICAL SOCIETY

**The best technical content in  
electrochemistry and solid state  
science and technology!**

**Available until November 9, 2020.**

**PRIME™**  
PACIFIC RIM MEETING  
ON ELECTROCHEMICAL  
AND SOLID STATE SCIENCE  
**2020**

**REGISTER TO ACCESS  
CONTENT FOR FREE! ▶**

# Implementation of the nonlinear programming algorithm when optimizing the final turning mode

**M G Galkin and A S Smagin**

Ural Federal University, 19, Mira street, Ekaterinburg 620000, Russia

E-mail: a.s.smagin@mail.ru

**Abstract.** For computer-aided multi-variant design of machining technologies, it is important to optimize the cutting parameters at the final pass in each technological operation. When carrying out designing procedures, there emerge problems relating to the algorithm of choosing the decision-making method, the objective function, and the regions of feasibility at final machining steps. Linear programming is time-consuming for multi-variant and multi-pass machining, if the algorithm is to be clear. It is known that when simulating the optimal metal-cutting process, the optimization criterion and the system of constraints are non-linear. Therefore, a computational algorithm can be made significantly more efficient if it is a non-linear algorithm based on Lagrange multipliers. Such approach to design helps simplify automating the computational algorithm for multi-pass single-tool machining with a precision cutting tool (a reamer). This is the method discussed herein.

## 1. Introduction

Selecting optimal cutting parameters is crucial for designing the machining process. It helps develop high-quality technological solutions. Large-scale application of this mechanism is limited due to the necessity of developing a complex mathematical model for the cutting process as well as due to the resource-intensiveness of the computational algorithm needed to design a large number of passes. Therefore, manual design is virtually impossible in such cases. Using a computer-aided computational algorithm makes the process more efficient, thus more practical.

Apparently, one may use different optimization criteria for such problems. The basic criteria consist in maximizing the performance or minimizing the cost of workpiece machining. Therefore, choosing the right objective function is an important prerequisite of solving any optimization-related problem [1], which particularly applies to choosing the cutting parameters for reaming both through-holes and blind holes.

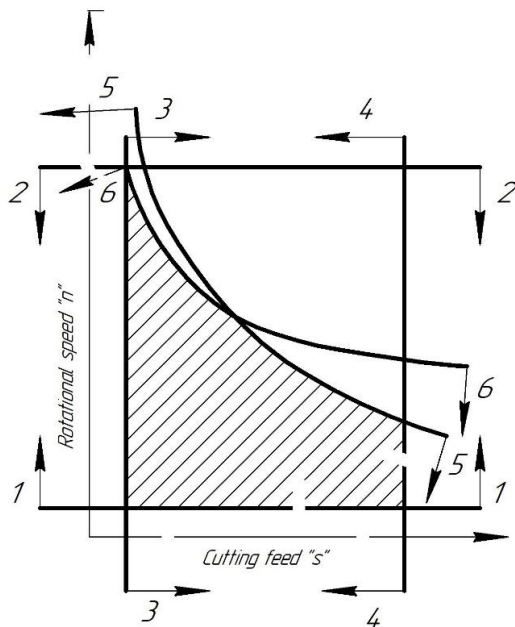
Professor G.K. Goransky was among the first researchers to propose a method for mathematical modeling of this process. In his interpretation, the model was a system of linear equations and constraints written as inequalities. Besides, the model contained a linear estimating function. The inequality system and the estimating function were derived by logarithmizing the corresponding equations describing this non-linear cutting process. The spindle rotation speed and the cutter feed-per-rotation were the parameters to optimize. It is these parameters that minimize the labor-intensiveness of workpiece machining. He further sought to optimize the parameters by using a graphical algorithm to find the optimal node [1]. This process complicated the computational model and necessitated multiple hand-run interim calculations.



## 2. Mathematical model of the cutting process

This paper proposes optimizing the spindle rotation speed and the reamer feed. In this case, the mathematical model is determined by how the parameters to optimize are functionally related to the requirements that condition the constraints imposed by the system at each specific machining stage. In paper [2] by Prof. G.K. Goransky, it is proposed to refer to thirteen factors affecting the cutting process. However, in a real-world setting a mathematical model that describes finishing the holes pre-machined with a drill or a with a countersink might actually contain a lot less constraints for modeling the process.

If we similarly synthesize a model for elementary passes done by a drilling or a reaming machine, it is rational to analyze six constraints at max. These are: minimum spindle rotation speed 1, maximum spindle rotation speed 2, minimum reamer feed 3, actual reamer feed (depends on the hole size and the hardness of the workpiece material) 4, reamer durability 5, and machine engine power 6. These constraints are graphically shown in figure 1. The hatched area is the feasible region.



**Figure 1.** Constraints for machining with a precision cutting tool.

The graphic dependencies show that the objective function and the constraint system are non-linear. Therefore, the mathematical model is going to be a non-linear optimization problem, one of those soluble by constrained optimization [3]. This method can be analytically written as the well-known equation system (1) [4]:

$$\begin{cases} F = f(x_i) \rightarrow \max(\min); \\ q_j(x_i) \leq a_j; \\ q_j(x_i) \geq b_j; \\ z_i \leq x_i \leq Z_i; \\ i = 1, n; j = 1, m. \end{cases}, \quad (1)$$

where  $F$  – is the target function or optimization criterion;  $q_j(x_i) \leq a_j$ ;  $q_j(x_i) \geq b_j$  – are constraints inter-variable dependencies.

It is known that this model (1) can be solved by using Lagrange multipliers [3]. This approach excludes logarithmizing the equations to subsequently write them as linear equations, which simplifies the computational algorithm. The above constraints applicable to machining with a precision cutting tool can be formalized as follows.

1. For the minimum spindle rotation speed:

$$n \geq n_{\min} = c_1. \quad (2)$$

2. For the maximum spindle rotation speed:

$$n \leq n_{\max} = c_2. \quad (3)$$

3. For the minimum reamer feed:

$$s \geq s_{\min} = c_3. \quad (4)$$

4. For the operating reamer feed:

$$s_{tab} \leq f(d_{hole}, \sigma_b, z_{nom}) = c_4. \quad (5)$$

5. For the necessary cutter durability:

$$n \cdot s^y_v \leq \frac{318 \cdot C_v \cdot D^{(q_v-1)} \cdot K_v}{T_n^m \cdot t^x_v} = c_5. \quad (6)$$

6. For the primary-motion engine power constraint:

$$n \cdot s^y \leq \left( \frac{19,5 \cdot 10^5 \cdot N_{dv} \cdot \eta}{C_p \cdot D_{hole} \cdot t^x} \right) = c_6. \quad (7)$$

The dependencies forming such inequalities are synthesized per the formulas described in papers [5], [6], [7], [8]. In equation (7),  $C_p$  and all the coefficients in the power value are adopted from the data for straight-turning tools [6].

It makes sense to assume the maximum feed per minute as the optimization criterion, which determines the minimum primary machining time for each  $i$ th surface. Thus, the three-parameter problem can be conditionally reduced to a two-parameter one. The cutting depth  $t$ , a third cutting parameter, is constrained by the minimum stock to be removed.

Therefore, the objective function can be written as:

$$F = n \times s \rightarrow \max. \quad (8)$$

Substitute the system of constraints and the target function (8) in the dependency (1) to synthesize the following equation system:

$$\begin{cases} n \geq C_1; \\ n \leq C_2; \\ s \geq C_3; \\ s_{tab} \leq C_4; \\ n \cdot s^y_v \leq C_5; \\ n \cdot s^y \leq C_6. \end{cases} \quad (9)$$

### 3. MS Excel as a solver

This mathematical model can be efficiently solved by using the *Visual Basic for Application* programming environment built in the Ms Excel spreadsheet processor.

Leon Lasdon, University of Texas at Austin, and Allan Waren, Cleveland State University, were the first to implement a non-linear optimization algorithm using *MS Excel* [9], [10], [11]. Using VBA for this purpose helps simplify the code by using various ready-made applications for implementing this algorithm [12], [13].

To solve the problem, we have modeled a user dialog box to enter the source data and display the calculation results, see figure 2.

**Figure 2.** Blank dialog box for solving the optimization problem.

This dialog box visualizes source data input as kinematic machine parameters (Group 1), as parameters to be picked from the reamer feed database (Group 2), as the power parameters of the cutting process (Group 3), or as the kinematic parameters of the cutting process (Group 4). Group 5 implements a mechanism of choosing the optimal cutting-process parameters and feed-per-minute calculation to perform a pass.

At the second stage, we develop program code in the Visual Basic for Application programming environment [13]. It consists of three procedures. The first procedure, "start", visualizes the dialog box on the data input screen.

```
Sub start ()
    Sheets("deploy").Select
    DialogSheets("window_deploy").Show
    With DialogSheets("window_deploy")

        .DropDowns(1).RemoveAllItems
        .DropDowns(1).AddItem "10;15;20;25;30;45;60" – resistance
        -----
        .DropDowns(2).RemoveAllItems
```

```

.DropDowns(2).AddItem "Steel; Cast Iron; Copper " – material
-----
.DropDowns(3).RemoveAllItems
.DropDowns(3).AddItem "to_15; to_20; to_30" – diameter interval
-----
.DropDowns(4).RemoveAllItems
.DropDowns(4).AddItem "Rough; Finishing; Fine" – processing
-----
End With
End Sub

```

The second procedure, "poisk", assigns variable values from the dialog box and computes the constraints; the output is presented in the dialog box, see figure 2.

```

Sub poisk ()
With DialogSheets("window_deploy ")
'assigning variables their values from the dialog box
(stoi; dhole; nmin; nmax; smin; qv; yv; m; kv) = Val(.EditBoxes(i).Text)
(nd; kpd; cp; x; y; lovr) = Val(.EditBoxes(i).Text)
-----
'reamer feed selection and computation of constraints
interv_dr = .DropDowns(3).Text
mat = .DropDowns(2).Text
vid_ob = .DropDowns(4).Text
Cells(40, 4) = mat
Cells(40, 5) = vid_ob
Cells(40, 2) = interv_dr
VPR(mat; R21C2:R38C7; interv_dr)
so = Cells(41, 1) ' table-specified feed per reamer turn
c5 = (318 * cv * dhole^ (qv - 1) * kv) / (stoi ^ m * t ^ xv)
c6 = (1950000 * Ndv * kpd) / (cp * dhole * t^x)
'parameter transfer to the optimization model
Cells(9, 4) = nmin
Cells(10, 4) = nmax
Cells(12, 4) = smin
Cells(13, 4) = c5
Cells(14, 4) = c6
.Labels(25).Text = c5
.Labels(26).Text = c6
End With
End Sub

```

The third procedure, "Find\_Root", performs non-linear programming of the optimization model and shows the objective-function value in Group 5 of the dialog box [12], [14].

```

Sub Find_Root ()
' non-linear optimization module running
SolverOk
SetCell:= Cells(i, j), MaxMinVal:= 1, ValueOf:= 0, ByChange:= Range("B3:C3")
SolverSolve True
End Sub.

```

In this procedure, the standard SolverOk function, an MS Excel object-model application, solves the problem of programming the objective function and the constraint system non-linearly.

The function is structured as follows:

$$\text{SolverOK}(\text{SetCell}, \text{MaxMinVal}, \text{ValueOf}, \text{ByChange}) \quad (10)$$

Syntax of the dependency (10) is described as follows:

- 1) *SetCell* specifies the target cell in the *Ms Excel* object model.
- 2) *MaxMinVal* specifies that at the maximum objective-value function, the output is 1; at the minimum value, the output is 2; at a specific value, the output is 3.
- 3) *ValueOf* determines the value to which the objective-function value is further compared. If the previous parameter, *MaxMinVal*, equals 1 or 2, *ValueOf* equals zero.
- 4) *ByChange* specifies a cell range in the *Ms Excel* object model, where the optimal parameter values are to be found.

#### 4. The practical implementation

Consider reaming a 45-steel workpiece as a case for this method. The machined surface is 20 mm in diameter, 30 mm long, the tool will endure 30 minutes, the machining operation is finishing. Reaming is done by means of a 2H125JI vertical drilling machine.

Figure 3 shows the dialog box with the solution of this problem for optimal spindle rotation speed, optimum feed per rotation and feed per minute.

The dialog box in the figure shows that the optimum spindle rotation speed is 184 rpm, the optimum feed is – 0.88 mm/rotation or – 161.7 mm/min. With a machining length of 30 mm, the primary machining time equals 0.185 min. These parameters can be fixed in the machining chart.

SELECTION OF THE OPTIMAL MODE FOR DEPLOYMENT

THE INPUT DATA FOR THE SOLUTION OF THE OPTIMIZATION PROBLEM TO DEPLOY

Min. spindle rotation C1 45

Max. spindle rotation speed C2 2000

Min. reamer feed C3 0.1

Deployment feed selection window

Processed material Сталь

Diameter interval до25

Type of treatment Окончатель

Table feed sweep C4 0.88

Kinematic parameters

Resistance 30

Diameter 20

Depth 0.15

Coefficient Cv 10.8

Coefficient qv 0.3 Xv

Coefficient Yv 0.65 0.2

Coefficient m 0.4

RESTRICTION C5 169.13

Power parameters

Power 2.2 Kv 1.1 Kz no

efficiency 0.8 [Ps] no

For the moment For power Po

Coefficient Cm - Cp 20 no

Coefficient qm - qp 1

Coefficient Ym - Yp 0.75 no

Coefficient Xm - Xp 1 no

RESTRICTION C6 7150

RESTRICTION C7 not use

1\_Choosing a tabular flow and ratios

2\_The choice of limitations

WINDOW PROBLEM SOLVING

Spindle rotation frequency 183.7

Sweep feed per revolution 0.88

Minute sweep feed 161.7

Treatment length 30

3\_Processing mode optimization

Basic processing time 0.185

Figure 3. Optimization solution dialog box.

## 5. Conclusion

This computational model enables the user not only to optimize the machining parameters in a specified region, but also to considerably reduce the design time. Optimizing the cutting parameters for each pass in a multi-pass machining process is a fairly labor-intensive procedure.

## References

- [1] Shoup T 1979 *A practical guide to computer methods for engineers* (Prentice-Hall) p 255
- [2] Goransky G 1963 *Computer calculation of cutting regimes* (BSSR State Publ., Minsk) p 192
- [3] Bertsekas D 1987 *Constrained optimization and Lagrange multiplier methods* (Academic Press) p 395
- [4] Tsirlin A 2015 *Optimization methods for engineers* (Direct-Media Publ, Moscow) p 214
- [5] Guzeyev V, Batuyev V and Surkov I 2005 *Cutting regimes for turning and drilling-milling-and-boring machines with numerical control* (Mashinostroyeniye, Moscow) p 368
- [6] Kosilova A and Meshcheryakov R 1986 *Spravochnik tekhnologa-mashinostroitelya* vol 2 (Mashinostroenie Publ., Moscow) p 496
- [7] Kosilova A and Meshcheryakov R 2001 *Spravochnik tekhnologa-mashinostroitelya* vol 2 (Mashinostroenie Publ., Moscow) p 949
- [8] Matalin A A 2008 *Mechanical engineering technologies* (Mashinostroyeniye: Izd-vo Lan) p 512
- [9] Lasdon L 1970 *Optimization theory for Lange System* (Macmillan, New York) p 523
- [10] Busch-Vishnias J Pang and Lasdon L 2000 *Optical seursor desing using nonlinear programming* Engineering Optimization
- [11] Gajulapalli R and Lasdon L 2000 *Computational experience with a safeguarded barrier algorithm for sparse nonlinear programming* Journal of Computational Optimization and Applications
- [12] Korchak S N, Koshin A A, Rakovich A G and Sinitsyn B I 1988 *Computer-aided design systems for processes, tools, and cutters* (Mashinostroyeniye) p 352
- [13] Komyagin V 1996 *Programming in Excel on Visual Basic* (Radio i svyaz' Publ., Moscow) p 320
- [14] Kuritsky B Ya 1997 *Searching for optimal solutions in Excel* (St. Petersburg: BHV - St. Petersburg) p 384